

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

This Page Blank (uspto)



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 :

G06F 15/403

A1

(11) International Publication Number:

WO 93/12498

(43) International Publication Date:

24 June 1993 (24.06.93)

(21) International Application Number: PCT/EP92/02682

(22) International Filing Date: 21 November 1992 (21.11.92)

(30) Priority data:

805,447

9 December 1991 (09.12.91) US

(71) Applicant (for DE only): IBM DEUTSCHLAND GMBH
[DE/DE]; Pascalstraße 100, D-7000 Stuttgart 80 (DE).(71) Applicant (for all designated States except DE): INTERNATIONAL BUSINESS MACHINES CORPORATION
[US/US]; Armonk, NY 10504 (US).(72) Inventors: HOWELL, William, Edward ; 8165 Precinct
Line Road, North Richland Hills, TX 76180 (US). RED-
DY, Hari, Nukalapati ; 2100 Grayson Drive, Apt. 1422,
Grapevine, TX 76051 (US). WANG, Diana, S. ; 13
Creekmore Drive, Trophy Club, TX 76262 (US).(74) Agent: SCHÄFER, Wolfgang; IBM Deutschland Infor-
mationssysteme GmbH, Patentwesen & Urheberrecht,
Pascalstrasse 100, D-7000 Stuttgart 80 (DE).(81) Designated States: CS, DE, HU, PL, RU, UA, European
patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT,
LU, MC, NL, SE).

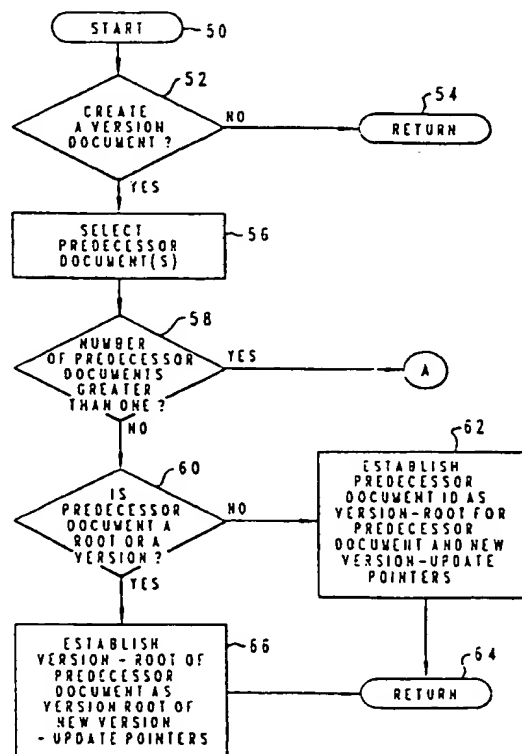
Published

With international search report.

(54) Title: METHOD AND SYSTEM FOR CREATING AND MAINTAINING MULTIPLE DOCUMENT VERSIONS IN
A DATA PROCESSING SYSTEM LIBRARY

(57) Abstract

A method and system for creating and maintaining multiple document versions in a data processing system implemented library. Selected documents within a data processing system implemented library are uniquely identified as root documents and a version-root identifier for each successor version of a particular root document is established. Thereafter, a selected version-root identifier is automatically associated with each created successor version of a root document. Upon the creation of a successor version of a root document and a second predecessor document, the version-root identifier associated with the root document is automatically associated with the successor version and the second predecessor document. Upon the creation of a successor version base upon multiple non-root predecessor documents a particular non-root predecessor document is automatically selected as a root document and a version-root identifier is established and associated therewith. Thereafter, the newly established version-root identifier is automatically associated with all non-root predecessor documents and the successor version based thereon.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FR	France	MR	Mauritania
AU	Australia	GA	Gabon	MW	Malawi
BB	Barbados	GB	United Kingdom	NL	Netherlands
BE	Belgium	GN	Guinea	NO	Norway
BF	Burkina Faso	GR	Greece	NZ	New Zealand
BG	Bulgaria	HU	Hungary	PL	Poland
BJ	Benin	IE	Ireland	PT	Portugal
BR	Brazil	IT	Italy	RO	Romania
CA	Canada	JP	Japan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SK	Slovak Republic
CI	Côte d'Ivoire	LI	Liechtenstein	SN	Senegal
CM	Cameroon	LK	Sri Lanka	SU	Soviet Union
CS	Czechoslovakia	LU	Luxembourg	TD	Chad
CZ	Czech Republic	MC	Monaco	TG	Togo
DE	Germany	MG	Madagascar	UA	Ukraine
DK	Denmark	ML	Mali	US	United States of America
ES	Spain	MN	Mongolia	VN	Viet Nam
FI	Finland				

D E S C R I P T I O N

METHOD AND SYSTEM FOR CREATING AND MAINTAINING
MULTIPLE DOCUMENT VERSIONS IN A DATA PROCESSING SYSTEM
LIBRARY

BACKGROUND OF THE INVENTION

1. Field of the Invention:

The present invention relates in general to data processing system implemented libraries and in particular to a method and system for creating and maintaining multiple document versions within a data processing system implemented library. Still more particularly the present invention relates a method and system for creating and maintaining multiple predecessor and successor versions of a document within a data processing system implemented library.

2. Description of the Prior Art:

Electronically implemented "libraries" are rapidly surpassing more traditional forms of information storage in the world today. It is quite common for data processing system implemented libraries to maintain and control thousand of different objects, or documents. The rules by which such documents may be altered and/or maintained are typically governed by various standards such as the International Standard ISO/IEC 10166, Document Filing and Retrieval (DFR).

One problem which exist in such electronic libraries is the management and control of multiple versions of a selected document in a so-called "Conceptual Document"

wherein all document therein are related. Many systems exist for governing relationships between multiple versions of a single document in a linear relationship. That is, a relationship wherein a selected document may have single predecessor document and a single successor document. While the need for a method and system which permits each document to have multiple predecessors and successor documents has been generally alluded to, no current library system permits such a relationship to be created and maintained.

It should therefore be apparent that a need exist for a method and system which permits multiple predecessor and successor versions of a document to be created and maintained within a data processing system library.

SUMMARY OF THE INVENTION

It is therefore one object of the present invention to provide an improved data processing system implemented library.

It is another object of the present invention to provide an improved method and system for creating and maintaining multiple document versions within a data processing system implemented library.

It is yet another object of the present invention to provide an improved method and system for creating and maintaining multiple predecessor and successor versions of a document within a data processing system implemented library.

The foregoing objects are achieved as is now described. Selected documents within a data processing system

implemented library are uniquely identified as root documents and a version-root identifier for each successor version of a particular root document is established. Thereafter, a selected version-root identifier is automatically associated with each created successor version of a root document. Upon the creation of a successor version of a root document and a second predecessor document, the version-root identifier associated with the root document is automatically associated with the successor version and the second predecessor document. Upon the creation of a successor version based upon multiple non-root predecessor documents a particular non-root predecessor document is automatically selected as a root document and a version-root identifier is established and associated therewith. Thereafter, the newly established version-root identifier is automatically associated with all non-root predecessor documents and the successor version based thereon.

BRIEF DESCRIPTION OF THE DRAWINGS

The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objects and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

Fig. 1 is a pictorial representation of a distributed data processing system which may be utilized to implement the method and system of the present invention;

- Fig. 2 is a high level logic flow chart illustrating a software implementation of the method and system of the present invention;
- Fig. 3 is a pictorial representation of a first Conceptual Document structure which may be implemented utilizing the method and system of the present invention;
- Fig. 4 is a pictorial representation of a second Conceptual Document structure which may be implemented utilizing the method and system of the present invention; and
- Fig. 5 is a pictorial representation of a third Conceptual Document structure which may be implemented utilizing the method and system of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

With reference now to the figures and in particular with reference to Fig. 1, there is depicted a pictorial representation of a distributed data processing system 8 which may be utilized to implement the method and system of the present invention. As may be seen, distributed data processing system 8 may include a plurality of networks, such as Local Area Networks (LAN) 10 and 32, each of which preferably includes a plurality of individual computers 12 and 30 respectively. Of course, those skilled in the art will appreciate that a plurality of Intelligent Work Stations (IWS) coupled to a host processor may be utilized to implement each such network.

As is common in such data processing systems, each individual computer may be coupled to a storage device 14 and/or a printer/output device 16. One or more such storage devices 14 may be utilized, in accordance with a method and system of the present invention, to store the various objects or documents which may be periodically accessed by a user having access to such objects or documents. Further, one or more such storage devices 14 may be utilized, as will be explained in greater detail herein, to store multiple predecessors and/or successors of a particular document within a data processing system implemented library, in accordance with a method and system of the present invention.

Still referring to Fig. 1, it may be seen that distributed data processing system 8 may also include main frame computers, such as main frame computer 18, which may be preferably coupled to Local Area Network (LAN) 10 by means of communication link 22. Main frame computer 18 may be coupled to a storage device 20 which may serve as remote storage for Local Area Network (LAN) 10 and may be coupled via communications controller 26 and communications link 34 to gateway server 28. Gateway server 28 is preferably an individual computer or Intelligent Work Station (IWS) which serves to link Local Area Network (LAN) 32 to Local Area Network (LAN) 10.

As discussed above with respect to Local Area Network (LAN) 32 and Local Area Network (LAN) 10, a plurality of data objects or documents may be stored within storage device 20 and controlled by main frame computer 18, as Resource Manager or Library Service for the data objects and documents thus stored. In the depicted

embodiment of the present invention such objects and/or documents are preferably stored and maintained in accordance with International Standard ISO/IEC 10166, Document Filing and Retrieval (DFR). Those skilled in the art will appreciate that it is often desirable to maintain a data processing system implemented library at a central location within a distributed data processing system wherein users throughout the distributed data processing system may access objects or documents stored therein.

Referring now to Fig. 2 there is depicted a high level logic flow chart illustrating a software implementation of the method and system of the present invention. As depicted, the process begins at block 50 thereafter passes to block 52 which illustrates a determination of whether or not the user desires to create a version of an existing document. If not, the process passes to block 54 and returns. In the event the user does desire to create a version of an existing document, as determined at block 52, the process passes to block 56. Block 56 illustrates the selection by the user of at least one predecessor document.

Next, the process passes to block 58 which, in accordance with an important feature of the present invention, illustrates a determination of the number of predecessor documents selected by a user. In the event a single predecessor document has been selected, the process passes to block 60. Block 60 depicts a determination of whether or not the predecessor document selected is a root document or a version of a root document. If not, the process passes to block 62 which illustrates the establishment of a version-root identifier for the predecessor document and the

- 7 -

association of that version-root identifier with both the predecessor document and the current document under creation. Thereafter, the pointers within the electronic library are updated and the process passes to block 64 and returns.

Referring again to block 60, in the event the predecessor document of the current version under creation is a root document or a version of a root document, the process passes to block 66. Block 66 illustrates the copying of the version-root identifier of the predecessor document to the current document under creation and the updating of the previous and next-version pointers for both documents. The process then passes to block 64 and returns.

Referring again to block 58 in the event the user has selected multiple predecessor documents for utilization in creation of the current version, the process passes to block 70. Block 70 depicts a determination of whether or not all predecessor documents selected by the user are either a root document or versions of a root document. If so, the process passes to block 72 which depicts a determination of whether or not all version-root identifications associated with the selected predecessor documents are identical. If not, the process passes to block 74 and returns an error message. Those skilled in the art will appreciate that in order to maintain a Conceptual Document wherein all documents within the Conceptual Document are related it is necessary that all documents contained therein have associated therewith an identical version-root identifier.

Referring again to block 72, in the event all predecessor documents selected by the user include

identical version-root identifiers, the process passes to block 76. Block 76 illustrates the assigning of the version-root identifier to the current document under creation and the updating of the pointers to those documents within the electronic library. The process then passes to block 78 and returns.

Referring again to block 70, in the event all predecessor documents selected by the user for inclusion in the creation of the present version document are not either root document or a version of a root document the process passes to block 80. Block 80 illustrates a determination of whether or not multiple predecessor documents within the plurality of predecessor documents selected by their user are either a root document or a version of a root document. If so, the process passes to block 82. Block 82 illustrates a determination of whether or not each of the multiple predecessor documents which is either a root document or a version of a root document includes the identical version-root identifier. If not, as above, the process passes to block 84 and returns an error message to the user.

In the event each of the multiple predecessor documents which is a document version includes an identical version-root identifier, as determined at block 82, the process then passes to block 86 wherein the version-root identifier present within the multiple predecessor documents is assigned to the current version under creation and all non-version predecessor documents within the plurality of predecessor documents selected by the user. The pointers to those documents are then updated. Thereafter, the process passes to block 88 and returns.

Referring again to block 80, in the event multiple predecessor documents within the predecessor documents selected by the user are not either root documents or versions of a root document, the process passes to block 90. Block 90 illustrates a determination of whether or not any single predecessor document within the multiple predecessor documents selected by the user is either a root document or a version of a root document and if so, the process passes to block 92. Block 92 illustrates the assigning of the version-root identifier of the single predecessor document which is either a root document or a version of a root document to the version under construction and all other non-version predecessor documents. Thereafter, the process returns, as depicted at block 94.

Referring again to block 90, in the event no single predecessor document within the plurality of predecessor documents selected by the user is either a root document or a version of a root document, the process passes to block 96. Block 96 illustrates an important feature of the present invention in which the system automatically selects one of the predecessor documents as a root document and thereafter assigns a version-root identifier of that document to all predecessor documents selected by the user and to the current version of the document under creation. The pointers to these documents are then updated and the process then returns, as depicted at block 98.

With reference now to Fig. 3, there is depicted a pictorial representation of a first Conceptual Document structure which may be implemented utilizing the method and system of the present invention. As illustrated, a

document C, at reference numeral 100, is depicted having multiple predecessor documents 102 and 104 and multiple successor documents 106 and 108. As described above with respect to the logic flowchart all version-root identifiers for predecessor documents 102 and 104, where present, must be identical and will thereafter be utilized as the version-root identifier for all successor documents within the Conceptual Document. In the event neither predecessor document 102 or predecessor document 104 includes a version-root attribute, the office Library Server, in accordance with an important feature of the present invention, will select the Unique Permanent Identifier of either predecessor document 102 or 104 as the version-root of the Conceptual Document depicted within Fig. 3 and this version-root identifier will then be added to the attribute of each document within the Conceptual Document depicted within Fig. 3. Thus, documents 100, 106 and 108 will each thereafter include a version-root identifier selected by the Library Server from either predecessor document 102 or predecessor document 104.

Referring now to Fig. 4 there is depicted a pictorial relationship of a second Conceptual Document structure which may be implemented utilizing the method and system of the present invention. As depicted in Fig. 4, document A, at reference numeral 110, is the predecessor document for document B, at reference numeral 112. Document B, at reference numeral 112, and document D, at reference numeral 116, are both predecessor documents of document C, at reference numeral 114. Thus, any document may be included in a Conceptual Document as a predecessor of another document which is being added to the Conceptual Document, if that document is not a member of a second Conceptual Document (there is no

version-root identifier present which is not identical to the version-root of the documents within the present Conceptual Document). In accordance with an important feature of the present invention, predecessor document D, at reference numeral 116, will automatically be assigned the same version-root identifier of document A, at reference numeral 110, as that version-root identifier will be automatically associated with document C, at reference numeral 114.

Finally, with reference to Fig. 5 there is depicted a pictorial representation of a third Conceptual Document structure which may be implemented utilizing the method and system of the present invention. As illustrated in Fig. 5, a predecessor document A, at reference numeral 120, is utilized to create a successor version document C, at reference numeral 122 and a successor version document B, at reference numeral 124. Document C also includes as a predecessor document B, at reference numeral 124. Thus, any document within the same Conceptual Document may have multiple predecessors, so long as those predecessors are not coincident. That is, document A, at reference numeral 120, and document B, at reference numeral 124, may be utilized as multiple predecessor documents for a third document since these predecessors share the same version-root identifier.

Additionally, the depicted embodiment of the present invention will automatically update all pointers within the library by automatically assigning a unique version-name to each newly created version document while also including in association with such version documents an identification of the immediate predecessor document(s). Further, when a document is utilized as a predecessor, an indication of the unique

version name of its successor will be associated with the predecessor document. In this manner, a Library Service may easily keep track of all related documents within a single Conceptual Document.

Upon reference to the foregoing those skilled in the art will appreciate that the Applicants herein have created a method and system which may be utilized to create and maintain multiple document versions in a data processing system implemented library wherein each document therein may have multiple predecessors and multiple successors. This flexible model of versioning allows many different combinations of Conceptual Document structures, greatly enhancing the flexibility of a data processing system implemented library.

While the invention has been shown in only one of its forms, it is not thus limited but is susceptible to various changes and modifications without departing from the spirit thereof.

C L A I M S

1. A method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library, said method comprising the data processing system implemented steps of:

uniquely identifying an original document within said library as a root document;

establishing a version-root identifier for all successor versions of said original document which may thereafter be created within said library;

automatically associating said version-root identifier with each successor document of said original document within said library; and

for each successor version of said original document and a second predecessor document, automatically associating said version-root identifier with said successor version and said second predecessor document wherein all related documents within said library will have an identical version-root identifier associated therewith.
2. The method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library according to Claim 1, further including the step of associating an identification of an immediate predecessor document with each successor document within said library.

3. The method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library according to Claim 1 or 2, further including the step of associating a unique version-name with each successor version of said original document.

4. A method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library, said method comprising the data processing system implemented steps of:

uniquely identifying a plurality of original documents within said library as root documents;

establishing a selected version-root identifier for all successor versions of each of said plurality of original documents which may thereafter be created within said library;

for each successor version of a uniquely identified one of said plurality of original documents and a second predecessor document, automatically associating said selected version-root identifier with said successor version and said second predecessor document;

for each successor version of a first predecessor document and a second predecessor document, automatically identifying a selected one of said predecessor documents as a root document and establishing a particular version-root identifier for all successor versions thereof; and

thereafter, automatically associating said particular version-root identifier with said successor version of said first predecessor document and said second predecessor document.

5. The method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library according to Claim 4, further including the step of associating an identification of an immediate predecessor document with each successor document within said library.
6. The method in a data processing system implemented library for creating and maintaining multiple versions of a document within said library according to Claim 4 or 5, further including the step of associating a unique version-name with each successor version of said original document.
7. A data processing system for creating and maintaining multiple versions of a document within a library within said data processing system, said data processing system comprising:

means for uniquely identifying an original document within said library as a root document;

means for establishing a version-root identifier for all successor versions of said original document which may thereafter be created within said library;

means for automatically associating said version-root identifier with each successor version of said original document within said library; and

means for automatically associating said version-root identifier with a successor version of said original document and a second predecessor document, and said second predecessor document, wherein all related documents within said library will have an identical version-root identifier associated therewith.

8. The data processing system for creating and maintaining multiple versions of a document within a library within said data processing system according to Claim 7, further including means for associating an identification of an immediate predecessor document with each successor version document within said library.
9. The data processing system for creating and maintaining multiple versions of a document within a library within said data processing system according to Claim 7 or 8, further including means for associating a unique version-name with each successor version of said original document.
10. A data processing program product for creating and maintaining multiple versions of a document within a library within a data processing system, said data processing program product comprising:

instruction means for uniquely identifying an original document within said library as a root document;

instruction means for establishing a version-root identifier for all successor versions of said original document which may thereafter be created within said library;

instruction means for automatically associating said version-root identifier with each successor version of said original document within said library; and

instruction means for automatically associating said version-root identifier with a successor version of said original document and a second predecessor document, and said second predecessor document, wherein all related documents within said library will have an identical version-root identifier associated therewith.

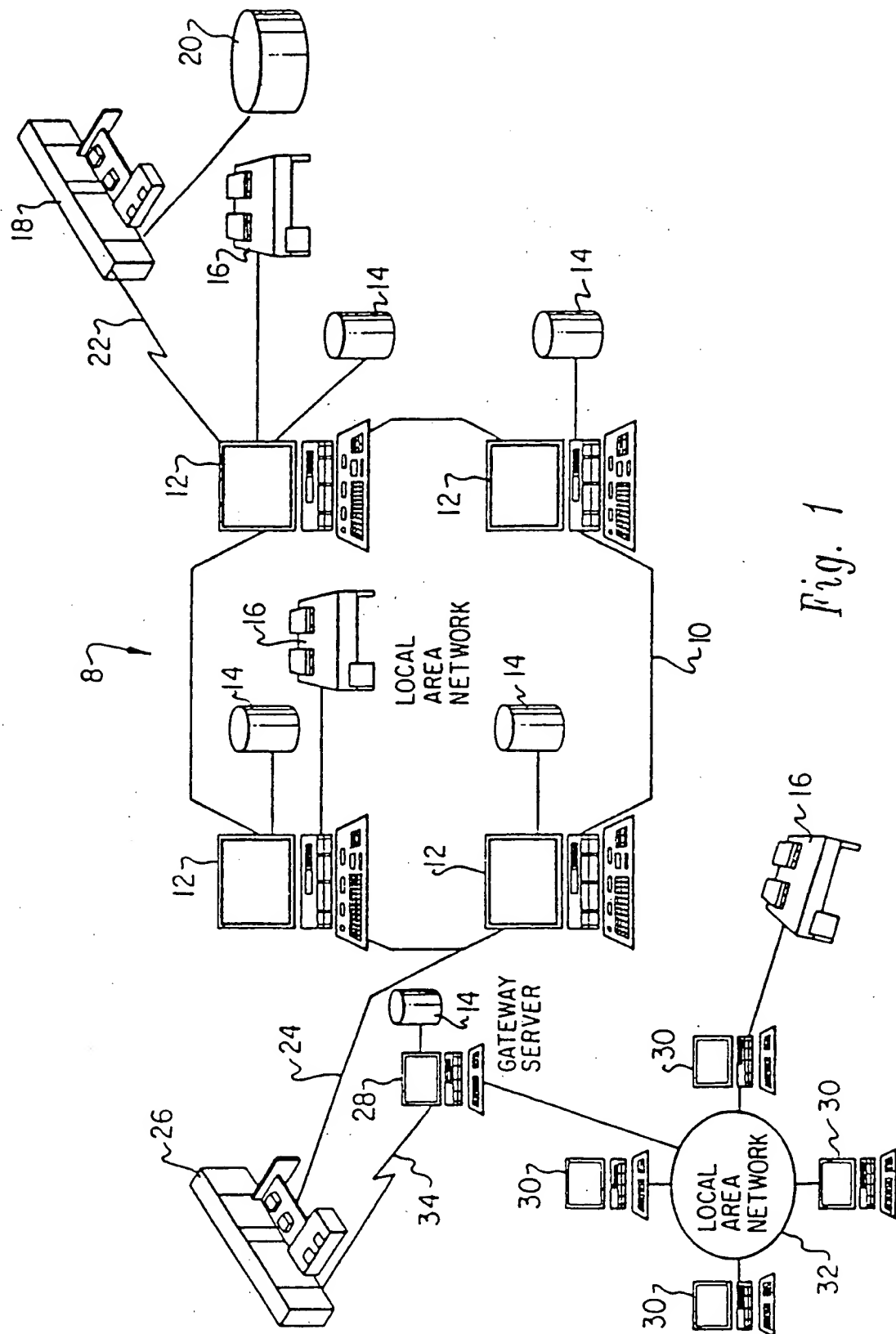


Fig. 1

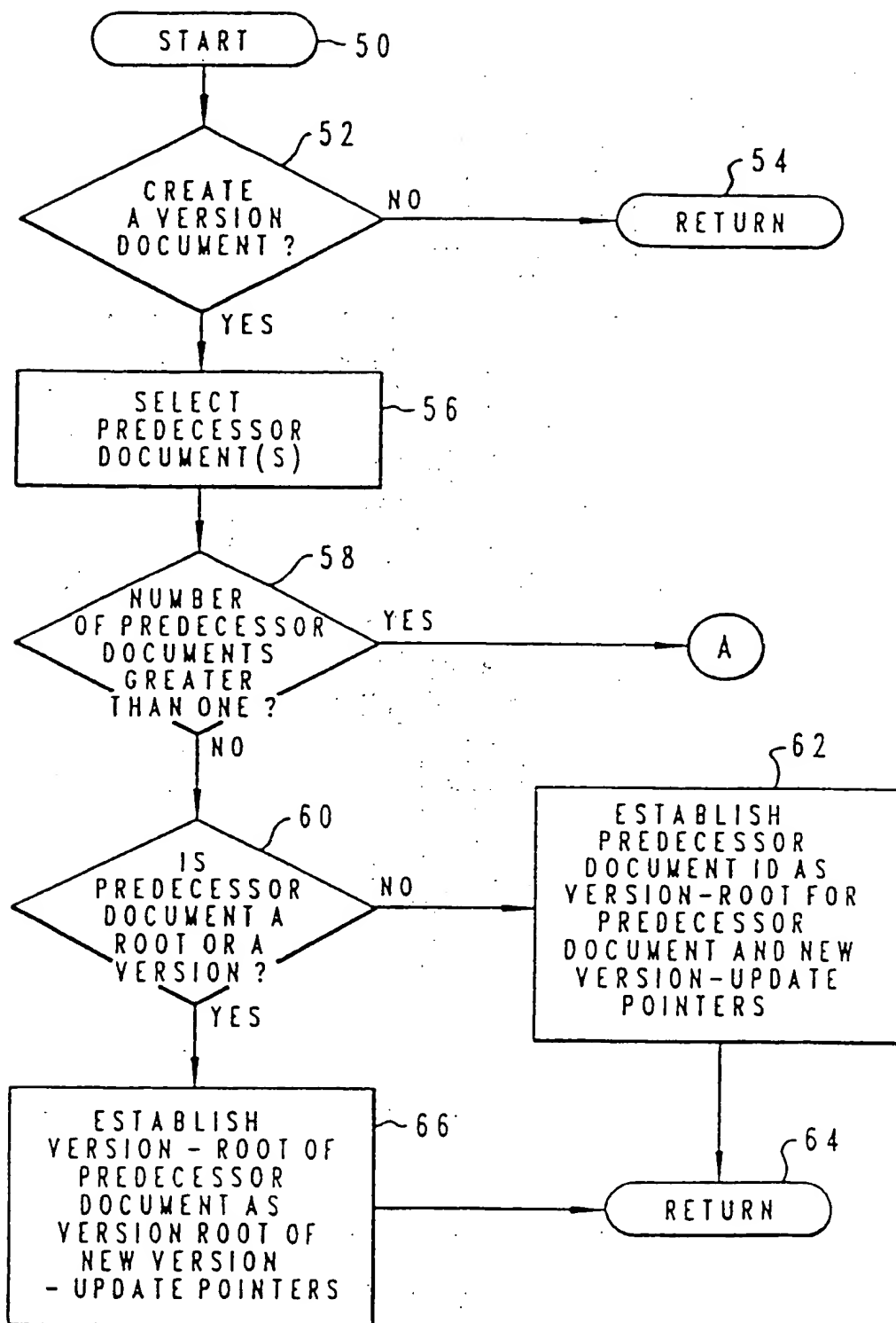


Fig. 2A

3/4

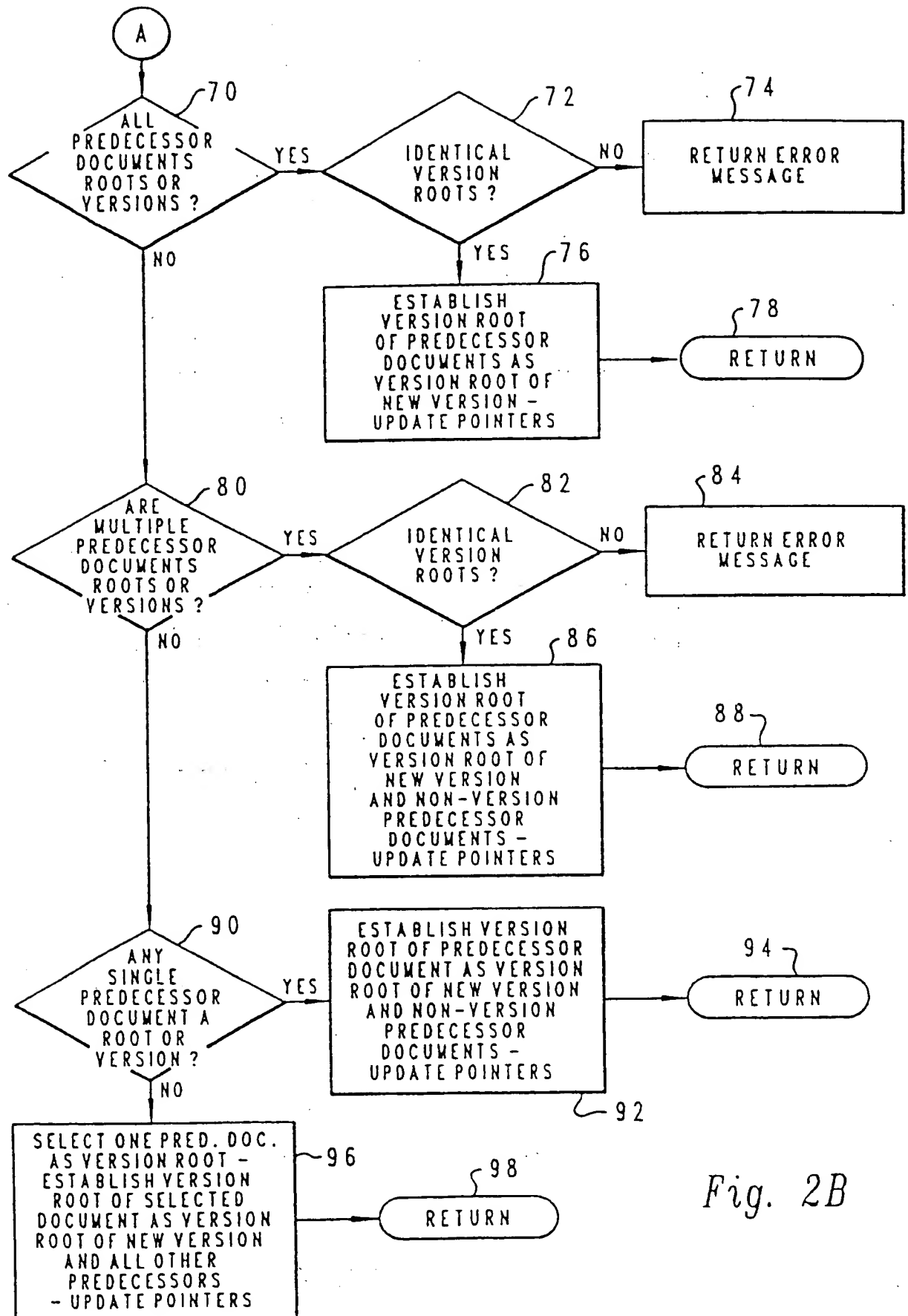
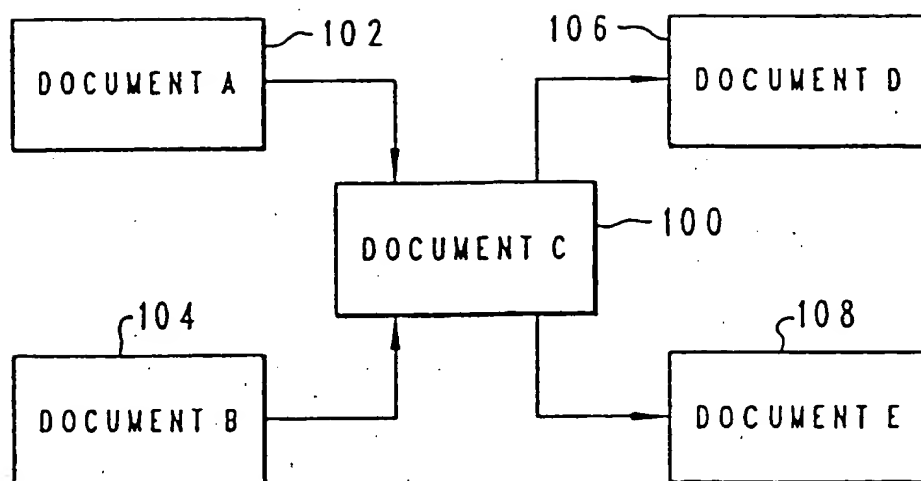
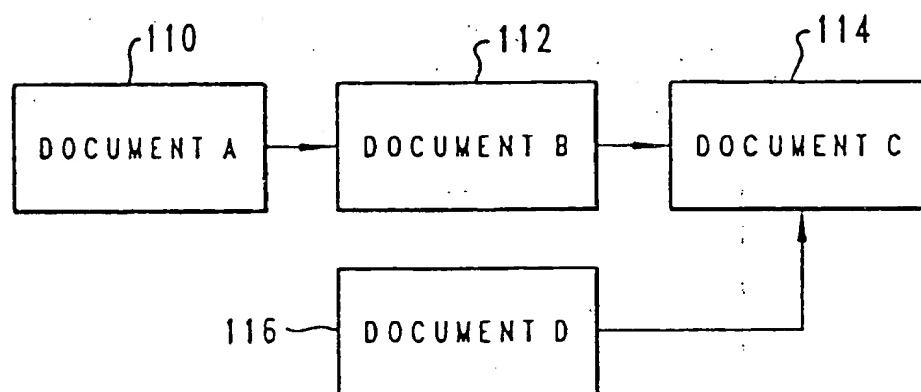
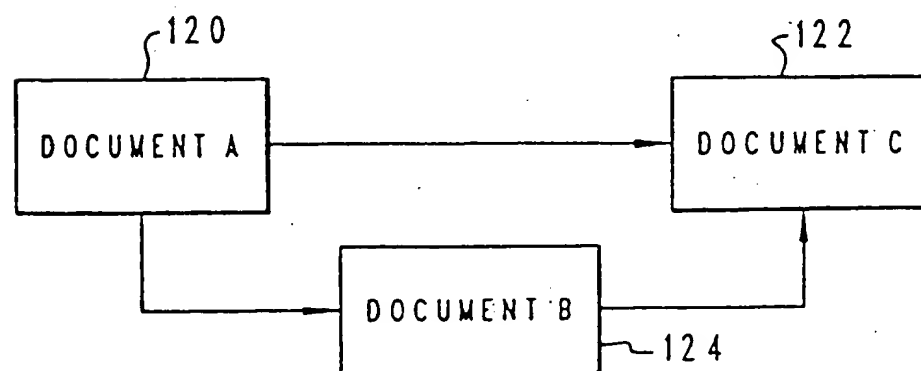


Fig. 2B

4/4

*Fig. 3**Fig. 4**Fig. 5*

INTERNATIONAL SEARCH REPORT

International Application No

PCT/EP 92/02682

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) ⁶		
According to International Patent Classification (IPC) or to both National Classification and IPC		
Int.Cl. 5 G06F15/403		
II. FIELDS SEARCHED		
Minimum Documentation Searched ⁷		
Classification System	Classification Symbols	
Int.Cl. 5	G06F	
Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched ⁸		
III. DOCUMENTS CONSIDERED TO BE RELEVANT⁹		
Category ¹⁰	Citation of Document, ¹¹ with indication, where appropriate, of the relevant passages ¹²	Relevant to Claim No. ¹³
X	PATENT ABSTRACTS OF JAPAN vol. 15, no. 79 (P-1170)25 February 1991 & JP,A,22 97 284 (HITACHI LTD.) 7 December 1990 see abstract ---	1,4,7,10
X	EP,A,0 230 616 (I.B.M. CORPORATION) 5 August 1987 see page 2, line 1 - page 2, line 45 ---	1,4,7,10
A	EP,A,0 371 606 (I.B.M. CORPORATION) 6 June 1990 see abstract; figure 2 -----	1,4,7,10
<p>¹⁰ Special categories of cited documents :</p> <p>"A" document defining the general state of the art which is not considered to be of particular relevance</p> <p>"E" earlier document but published on or after the international filing date</p> <p>"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>"O" document referring to an oral disclosure, use, exhibition or other means</p> <p>"P" document published prior to the international filing date but later than the priority date claimed</p> <p>"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention</p> <p>"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step</p> <p>"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.</p> <p>"&" document member of the same patent family</p>		
IV. CERTIFICATION		
Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report	
28 JANUARY 1993	05.02.93	
International Searching Authority	Signature of Authorized Officer	
EUROPEAN PATENT OFFICE	KATERBAU R.E.	

Form PCT/ISA/210 (second sheet) (January 1985)

**ANNEX TO THE INTERNATIONAL SEARCH REPORT
ON INTERNATIONAL PATENT APPLICATION NO.**

EP 9202682
SA 66736

This annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information. 28/01/93

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0230616	05-08-87	CA-A- 1267236 JP-A- 62173532	27-03-90 30-07-87
EP-A-0371606	06-06-90	JP-A- 2165377	26-06-90

EPO FORM P0477

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

This Page Blank (uspto)